

本节内容

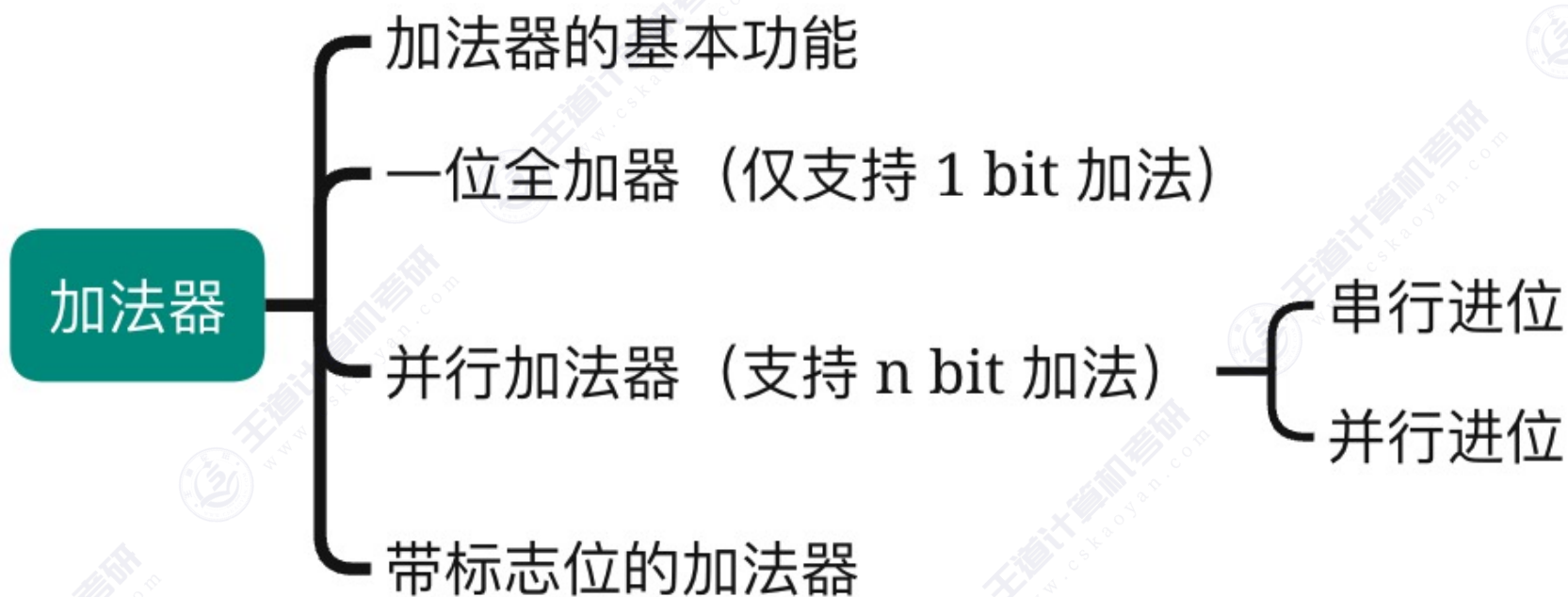
# 加法器

关注公众号【研途小时】获取后续课程完整更新！

王道24考研交流群：769832062

王道考研/CSKAOYAN.COM

## 本节总览



# 加法器的基本功能

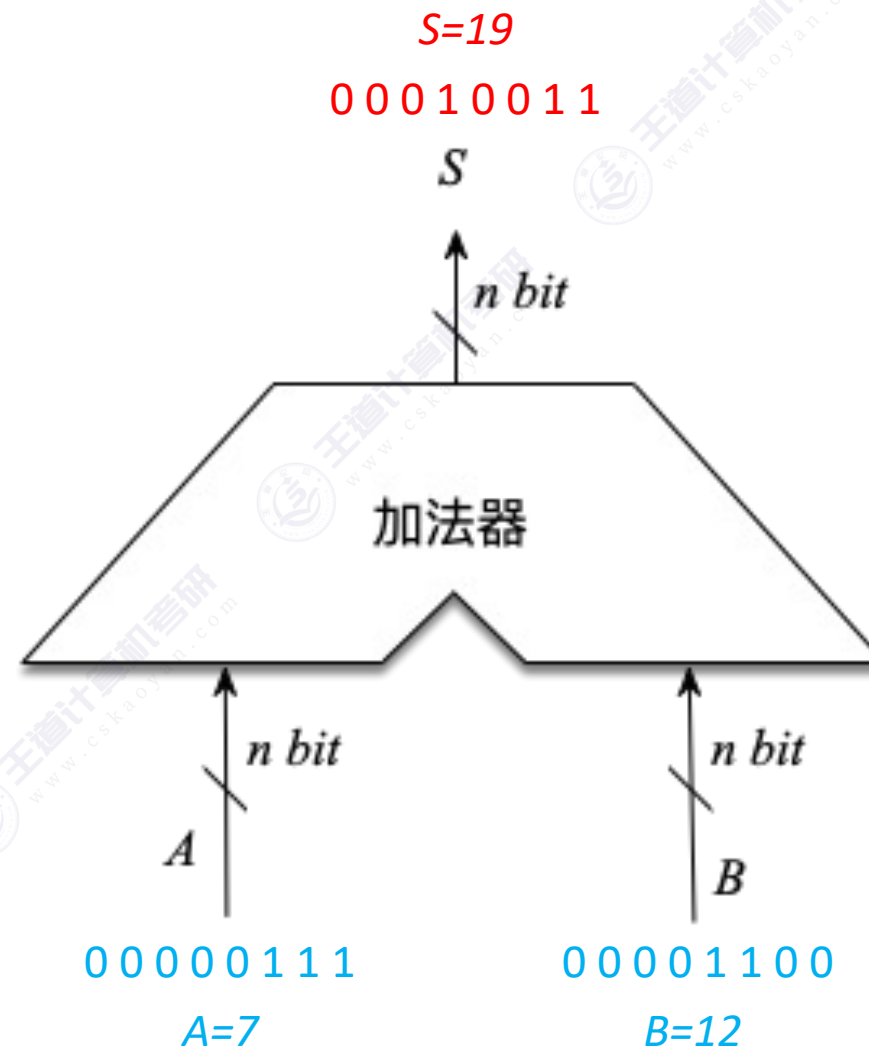
例：7+12=19

用8bit表示被加数A、加数B、和S



如果能用逻辑门电路实现一个 n bit 加法器

那么这件事情...



# 如何用门电路实现一位加法？

00000111  
+ 00001100  
-----  
00010011

向高位的进位

$C_i$

$A_i$  被加数的本位

$B_i$  加数的本位

$C_{i-1}$  来自低位的进位

$S_i$  本位和

输入

$A_i$

$B_i$

$C_{i-1}$

输出

$S_i$  : 输入中有奇数个1时输出1

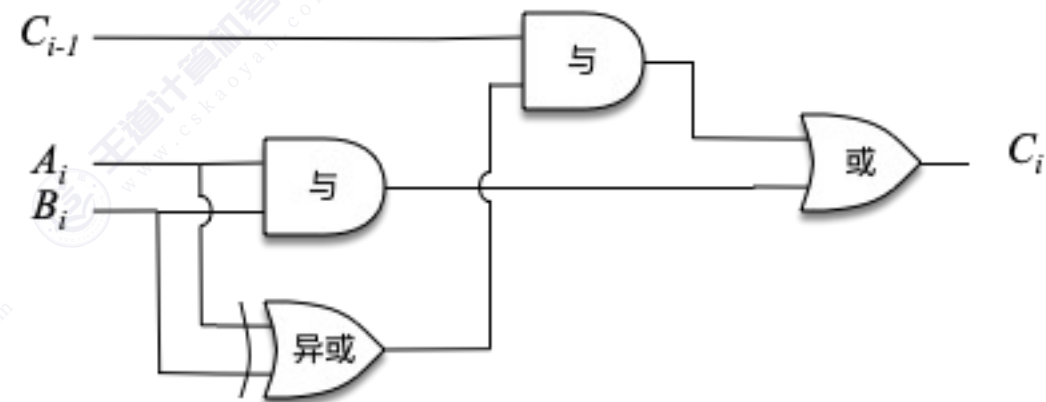
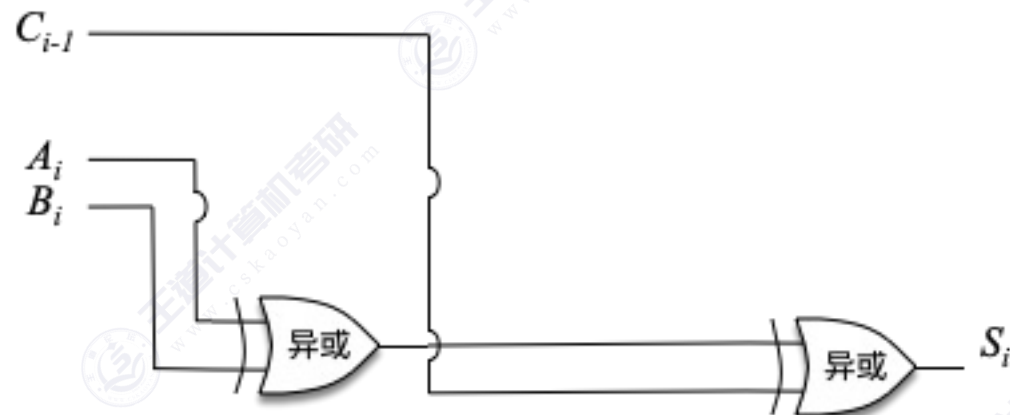
$$S_i = A_i \oplus B_i \oplus C_{i-1}$$

$C_i$  : 输入中至少2个1时输出1

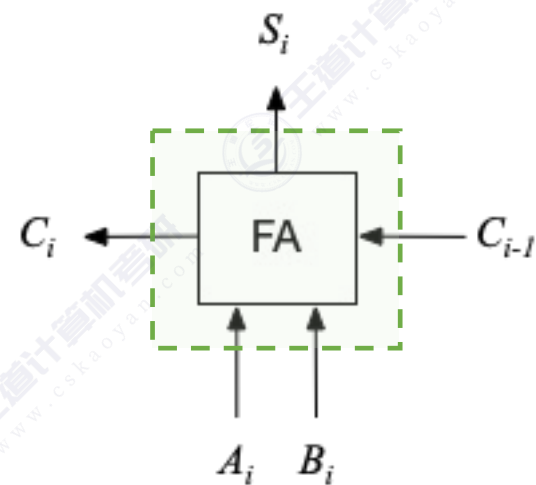
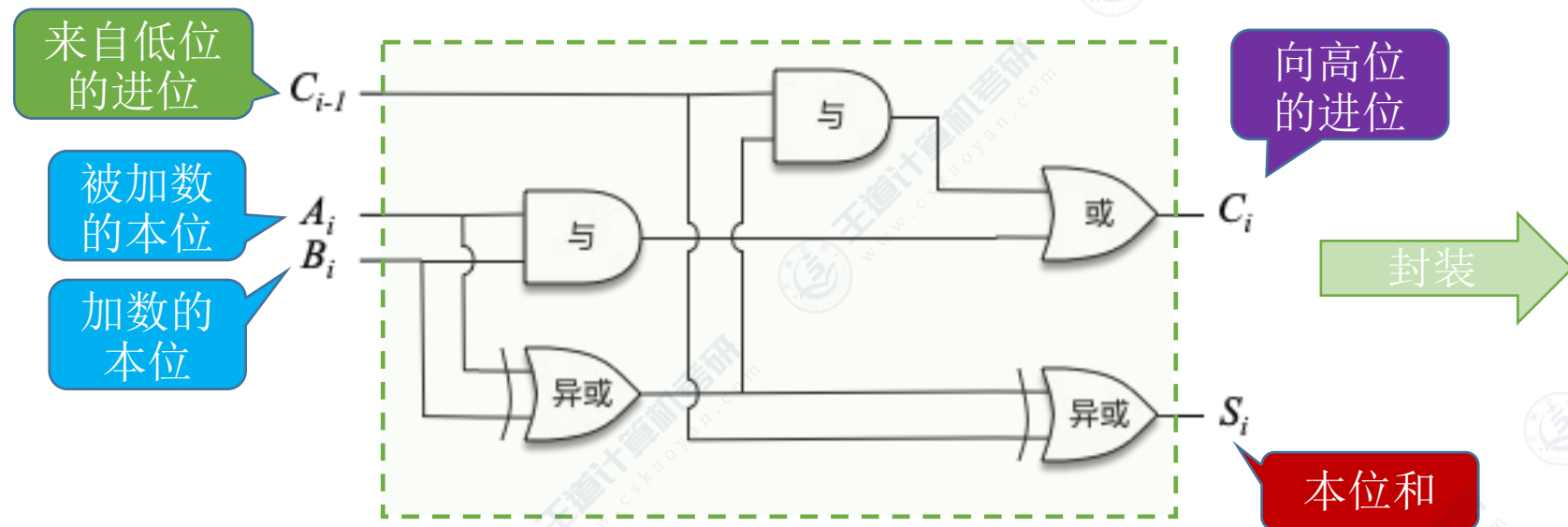
$$C_i = A_i B_i + (A_i \oplus B_i) C_{i-1}$$

两个本位都为1

两个本位中有一个1，且来自低位的进位是1



# 一位全加器



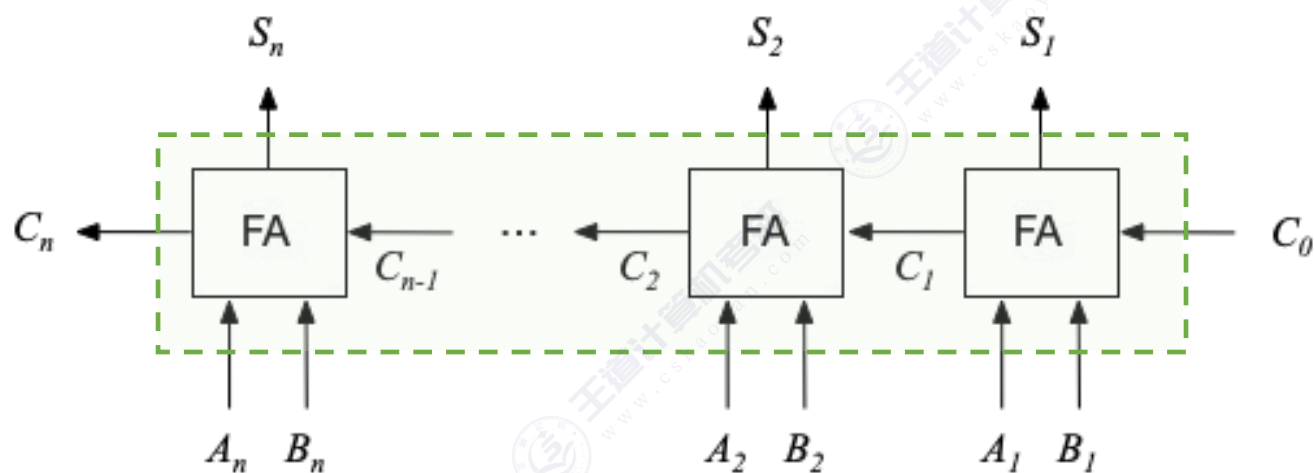
一位全加器 (FA, full adder)

$$S_i = A_i \oplus B_i \oplus C_{i-1}$$

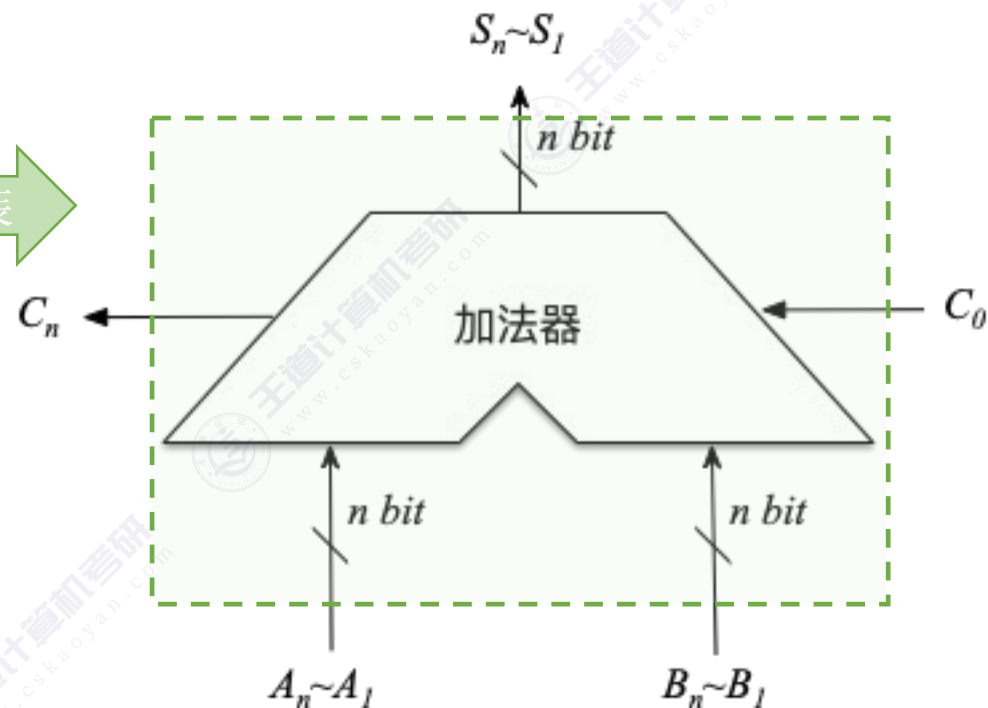
$$C_i = A_i B_i + (A_i \oplus B_i) C_{i-1}$$

# n bit 加法器

👉 把  $n$  个一位全加器串接起来，就可进行两个  $n$  bit 数的相加。



封装



$$\begin{array}{r}
 00000111 \\
 + 00001100 \\
 \hline
 00010011
 \end{array}$$

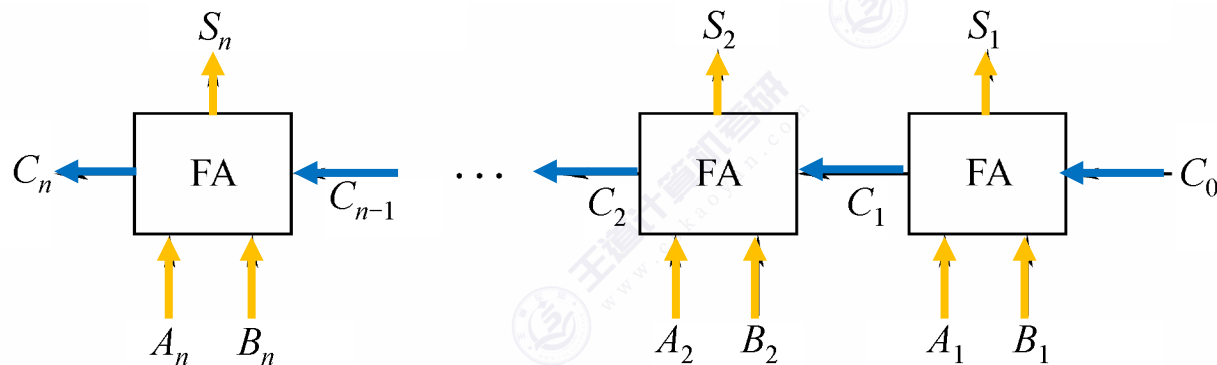
$A_i$

$B_i$

$C_{i-1}$

$S_i$

## 不足之处？



进位信息是串行产生的，计算速度取决于进位产生和传递的速度。位数越多，运算速度越慢。

注1：电信号到达稳态需要一定时间，因此进位产生速度会有延迟。

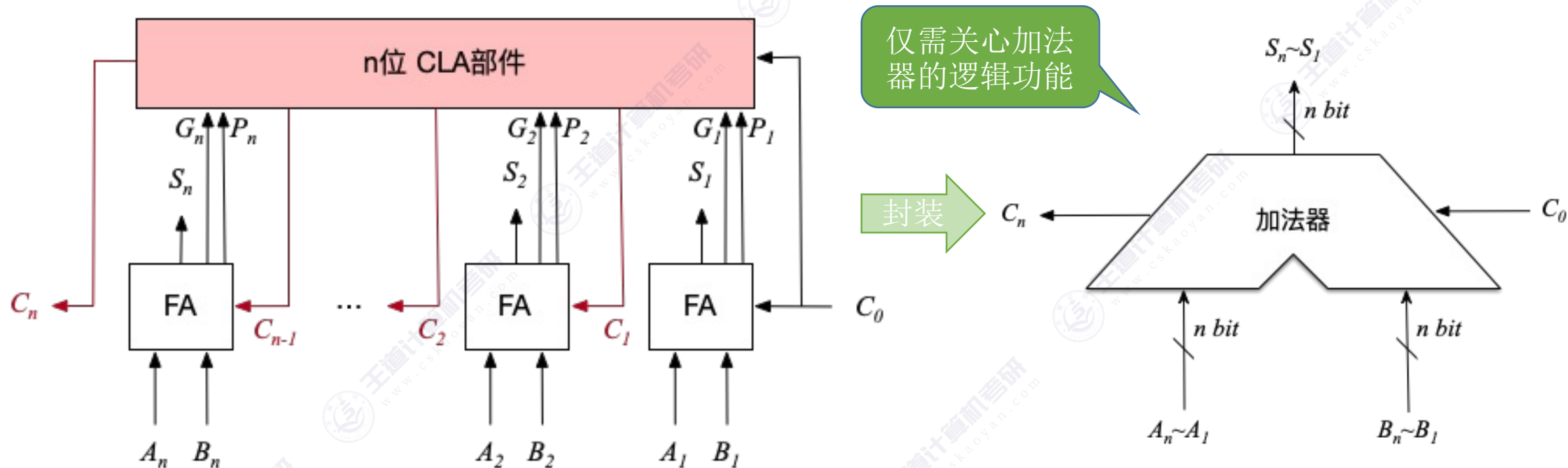
注2：串行进位又称为行波进位，每一级进位直接依赖于前一级的进位，即进位信号是逐级形成的。

由于两个输入端允许并行输入  $n$  bit，因此这种加法器属于：并行加法器

由于进位信息是串行产生的，因此从“进位方式”看，这种加法器属于：串行进位加法器

综上，很多教材把这种加法器称为“串行进位的并行加法器”

# 并行进位的并行加法器

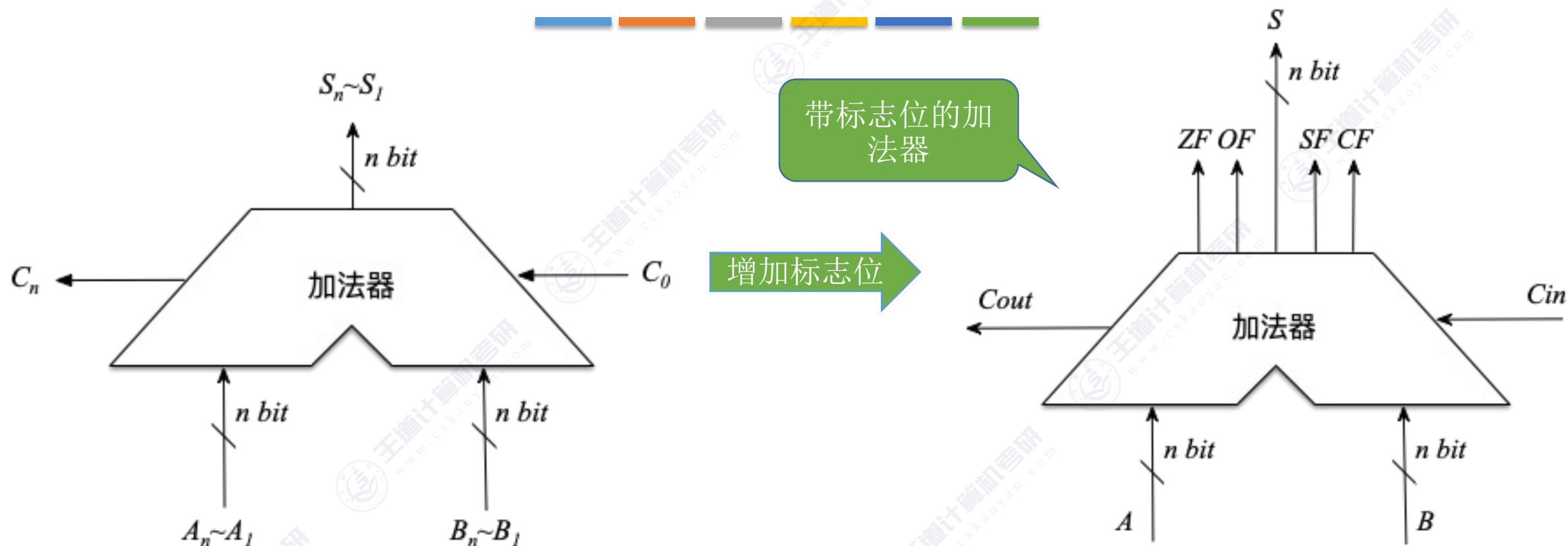


加入亿点点细节

**并行进位的并行加法器**：所有**进位信息都是同时产生的**，几乎没有延迟。  
特点：运算速度比“**串行进位的并行加法器**”更快。

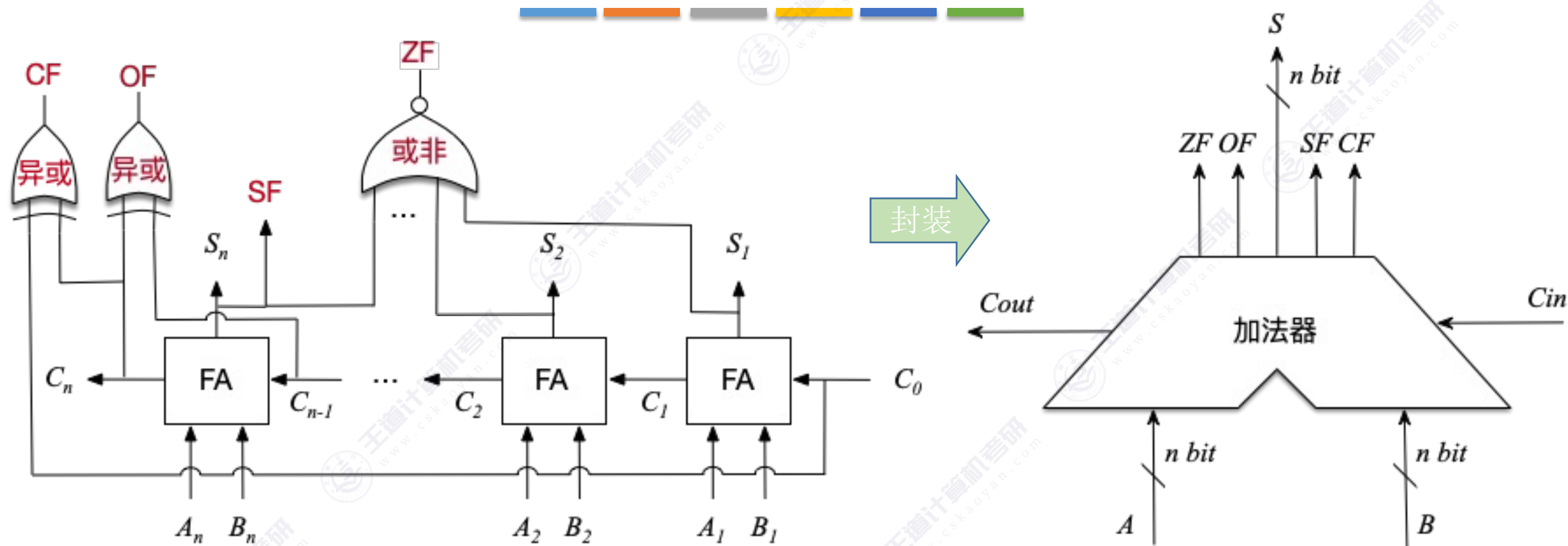


## 带标志位的加法器



OF (Overflow Flag) 溢出标志, 用于判断带符号数加减运算是否溢出。OF=1 溢出; OF=0 未溢出  
SF (Sign Flag) 符号标志, 用于判断带符号数加减运算结果的正负性。SF=1 结果为负; SF=0 结果为正  
ZF (Zero Flag) 零标志, 用于判断加减运算结果是否为0。ZF=1 表示结果为0; ZF=0 表示结果不为0  
CF (Carry Flag) 进位/借位标志, 用于判断无符号数加减运算是否溢出。CF=1 溢出; CF=0 未溢出

## 标志位的生成



$OF = C_n \oplus C_{n-1}$  —— 即最高位的进位  $\oplus$  次高位的进位。反映带符号数加减运算是否溢出。

$SF = S_n$  —— 也就是取运算结果的最高位（符号位）。反映带符号数加减运算的正负性。

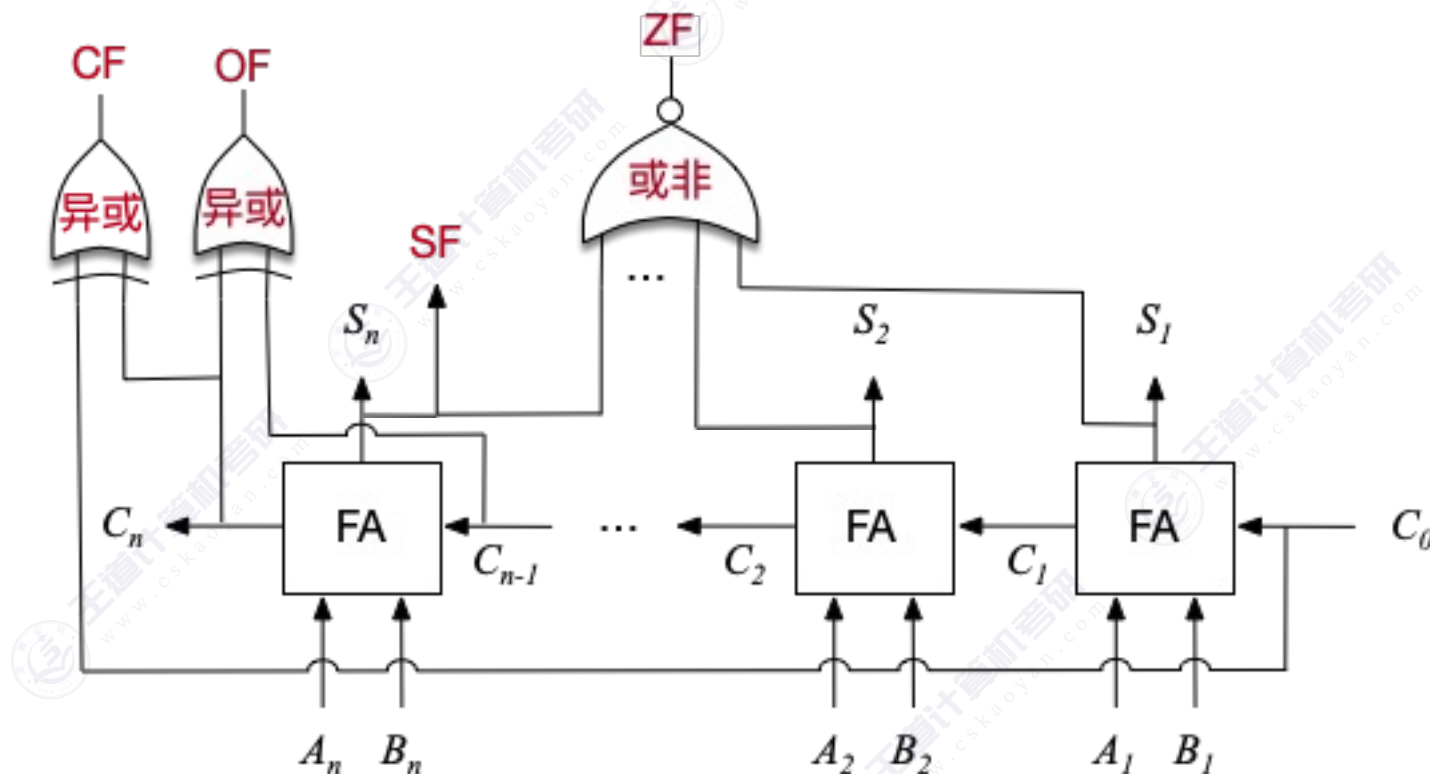
$ZF = \overline{S_n + \dots + S_2 + S_1}$  —— 仅当运算结果所有 bit 全0时，ZF才为1，此时表示运算结果为0。

$CF = C_{out} \oplus C_{in} = C_n \oplus C_0$  —— 反映无符号数加减运算是否溢出。

# 知识回顾



## 知识回顾



$OF = C_n \oplus C_{n-1}$  —— 即最高位的进位  $\oplus$  次高位的进位。反映带符号数加减运算是否溢出。

$SF = S_n$  —— 也就是取运算结果的最高位（符号位）。反映带符号数加减运算的正负性。

$ZF = \overline{S_n + \dots + S_2 + S_1}$  —— 仅当运算结果所有 bit 全0时，ZF才为1，此时表示运算结果为0。

$CF = C_{out} \oplus C_{in} = C_n \oplus C_0$  —— 反映无符号数加减运算是否溢出。



公众号：王道在线



b站：王道计算机教育



抖音：王道计算机考研